REMARKS/ARGUMENTS

Claims 1-19 are pending in this application.

Claim Rejections under 35 U.S.C. §112

Claims 4, 11 and 17 are rejected under 35 U.S.C. §112, first paragraph. Applicants request reconsideration of the rejection in view of the amendments made to the claims and for the following reasons. Applicants refer to a plurality of groups of the vacant storage regions. A group means one unit of storage when the access load is distributed to a plurality of storage devices 13 or LUs 131. See, page 10, lines 5-15 of the specification. Further, Applicants set forth divisions in the specification (312) as being created as a result of dividing a virtual volume 100 in order to disperse the access load (to the respective divisions). See, page 12, lines 4-7 of the specification. Accordingly, Applicants respectfully assert that claims 4, 11 and 17 are enabled by the specification within the meaning of 35 U.S.C. §112, first paragraph, and therefore the rejection should be withdrawn.

Claims 1, 4, 5, 8, 11, 12, 14, and 17-19 are rejected under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Applicants have amended the claims to overcome the rejection.

Claim Rejections under 35 U.S.C. §§102 and §103

Claims 1-3, 6-10, 13-16 and 19 are rejected under 35 U.S.C. §102(b) as being

anticipated by Blandy et al., U.S. Patent No. 5,390,315 (Blandy). Further, claims 4, 11 and 17 are rejected under 35 U.S.C §103(a) as being unpatentable over Blandy, U.S. Patent No. 5,390,315 in view of Obara et al., U.S. Patent No. 6,378,039. Applicants request reconsideration of the rejections for the following reasons.

According to the storage allocation method, virtualization device, program and system of the invention, a vacant storage region is allocated to a virtual volume from one or more storage devices by acquiring a storage region for allocation from the vacant storage region so that a remaining unallocated part of the required size to be allocated becomes smaller than a specified maximum region size of the vacant storage region. And, when the remaining unallocated part of the required size to be allocated becomes smaller than the maximum region size of the vacant storage region, a storage region having a size that is a smallest increment of the power of two that is not smaller than the remaining unallocated part of the required size is allocated from the vacant storage region. In the present invention, allocation efficiency is achieved. On the other hand, in the prior art, the allocation of a vacant region can become inefficient and wasteful, as explained in the Background of the Invention section of the specification.

Specifically, as described in US Patent No. 6,131,150, a technique called the buddy algorithm (abbreviated as B.A. in specification and in Figs. 5 and 6) has been used for allocating vacant regions, primarily to prevent fragmentation of vacant regions in the storage space or memory. According to this method, the vacant space of the memory is allocated in such a manner that the allocated memory size is always an increment of the power of two, and therefore, the allocated space is an increment of the power of two that is greater in size than the

required size to be allocated. While this method is useful for avoiding fragmentation, the size of the allocated region that results from implementation of the method can become excessively large as compared with the required size to be allocated. That is, the difference between the required size to be allocated and the smallest increment of the power of two exceeding the required size that is allocated can be large in some cases. If many relatively large real regions are allocated as a result, the overall allocation efficiency becomes substantially degraded.

A comparison of the prior art B.A. method and examples of allocating the vacant storage region according to the invention are explained with respect to the diagram of Fig. 7(c) of the application. Fig. 7(c) shows a required size to be allocated (real region size) of 150 GB. In the prior art B.A. method, the allocation of the vacant region would be set to one of an increment of the power of two, e.g. 128 GB, 256 GB, 512 GB, etc. Accordingly, the B.A. method would allocate a 256 GB region as the allocated region for a required size of 150 GB. The difference between the allocated region and the required size, therefore, would be 106 GB in this example, which would be wasteful, as explained in the Background of the Invention section of the specification.

According to the present invention, on the other hand, as set forth in the first embodiment of the invention (*see* the flow chart of Fig. 5), the first step of allocating the vacant space would refer to a specified maximum region size of the vacant space. *See*, page 13, lines 7-11 of the specification which defines the maximum region size as being an integer power of two, for example. In the example shown in Fig. 7(c), the maximum region size, which can be set by an administrator, is 32 GB. In the first step of the invention, the region to be allocated from the vacant storage region would be allocated by executing a loop in the flow chart of Fig.

5 that includes steps 2109, 2115, 2116, 2104, 2109, 2120 and 2111 through 2114, for example. This would result in allocating the maximum region size four times for a total of 128 GB. Then, the remaining unallocated part of the required size to be allocated would become smaller than the specified maximum region size of the vacant storage region, and this would be determined in step 2109 of Fig. 5. Following the flow chart of Fig. 5, a storage region having a size that is a smallest increment of the power of two that is not smaller than the remaining unallocated part of the required size would then be allocated using the B.A. method as shown in steps 2115 and 2116. In the example of Fig. 7(c), the remaining unallocated part of the required size to be allocated would be 22 GB and the smallest increment of the power of two that is not smaller than the remaining unallocated part of the required size would be 32 GB, for a total of 160 GB. According to the invention, therefore, a 160 GB region of memory would be allocated in the present invention as compared with a 256 GB allocation of memory in the B.A. method of the prior art, which results in allocation efficiency not appreciated by the prior art.

Blandy, on the other hand, is concerned with a technique for allocating locations on direct access storage drive DASD that efficiently uses available storage in the DASD and minimizes overhead associated with accessing the data on the DASD. According to Blandy, it is desirable to minimize the amount of radial movement of the read/write head required for each access, so the data to be written to the DASD should be stored in one contiguous set of locations wherein a single I/O operation suffices because only one start location and one length need be specified.

Specifically, in Blandy, the DASD is divided into sets of slots of a fixed size, i.e. contiguous storage slots (a "slot" being just large enough to store one page of data). The slot

allocation is accomplished by a slot allocation routine that searches for a set length that is greater than the requested allocation. The slot allocation routine then determines if the set length exceeds the requested allocation by a large fragment, i.e. a length that is likely to satisfy most subsequent requests (step 530 in Fig. 3). If the set length exceeds the allocation request by a large fragment, the first number of slots of the set which equal the allocation request are selected to satisfy the request (step 550). Then, the respective bit map is updated (step 552) to indicate that these slots are no longer available. Accordingly, Blandy does not disclose the first and second steps of the present invention that are implemented by the storage allocation method, virtualization device, program and system of the invention. Therefore, Blandy does not anticipate the invention as claimed by Applicants in claims 1-3, 6-10, 13-16 and 19, and the rejection under 35 U.S.C. § 102(b) should be withdrawn.

Claims 4, 11 and 17 are dependent claims that are rejected using the combination of Blandy and Obara et al., U.S. Patent No. 6,378,039. Obara is relied upon for disclosing a method for balancing loads across a plurality of disks, however the reference unable to overcome the deficiencies noted in the Blandy reference. Accordingly, claims 4, 11 and 17 should be allowed at least for depending from base claims asserted to be allowable for the foregoing reasons.

CONCLUSION

In view of the foregoing, Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

Respectfully submitted,

MATTINGLY, STANGER, MALUR & BRUNDIDGE, P.C.

By _____
John R. Mattingly
Reg. No. 30,293
(703) 684-1120

JRM/so
Date: March 2, 2006